Use of Evaluation Criteria in Security Education

Thuy D. Nguyen and Cynthia E. Irvine
Naval Postgraduate School, Monterey, CA, USA
tdnguyen@nps.edu
irvine@nps.edu

Abstract: Success in information warfare will depend on resilient, reconstitutable cyber assets and the ability to assess and respond to attacks. A cornerstone of this success will be the ability of Information Assurance professionals to develop sound security requirements and determine the suitability of evaluated security products for mission-specific systems. Recognizing the pedagogical value of applying security evaluation criteria such as the Common Criteria (CC) to information security education, we recently introduced a graduate-level Computer Science course focusing on methodical security requirements engineering based on the CC. This course aims to provide students with an understanding of how security evaluation criteria can be used to specify system security objectives, derive security requirements from security objectives, establish life cycle and development processes, and provide an organizational framework for research and development. Although imperfect, the paradigmatic process of the CC provides a usable framework for in-depth study of various tasks relating to system requirements derivation and verification activities: system requirements elicitation, threat analysis, security objectives definition and security requirements expression. In-class discussions address fundamental security design principles and disciplines for information and software assurance (e.g., formal methods and life cycle management) as applied to security requirements derivation. Coverage of advanced CC topics includes high assurance evaluation, international and U.S. scheme interpretation processes, guidance for Protection Profile development, CC evaluation methodology, and composite evaluation. This paper describes the scope and design of a pilot course offering. Laboratory projects focus on the differences between security functional and assurance requirements, mock evaluation of a draft Protection Profile, examination of the interpretation process in the U.S. scheme, and development of a preliminary sketch of a Composed Assurance Package for a hypothetical composed target system that is suitable for use in operational environments requiring medium robustness. Lessons learned and planned refinements of course material and focus are also discussed.

Keywords: Security requirements engineering, security evaluation, security education, information assurance

1. Introduction

The classic illustration of the misaligned ends of a railroad track with the two construction teams arguing over the blueprint sums up pointedly the importance of having correct and sound requirements prior to development. It is commonly understood that an effective way to avoid miscommunication is to utilize a systematic development process such that the correctness of the implementation (i.e., realization of abstract design) can be analyzed. An important part of this process is requirements engineering which concerns with capturing user's needs in a tractable form that can be used to guide development activities and analysis (Nuseibeh 2000).

Requirements engineering is inherently difficult because it deals with defining the problems to be solved. Other system and software engineering disciplines are easier since they relate to solving well-defined problems (Cheng 2007). Security further complicates the issue since it is imperative that requirements for a secure system are properly defined so that an assurance case can be made about the correctness of the system's security properties. The ability to derive correct security requirements requires critical understanding of key concepts and approaches to domain-specific problems. In keeping with the broader goal of preparing our graduates for real world challenges, the security track of the Computer Science (CS) department at our institution introduced a graduate level course that uses security evaluation criteria as a means to teach security requirements engineering.

Report Documentation Page

Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **2008**	2. REPORT TYPE	3. DATES COVERED **00-00-2008 to 00-00-2008**
4. TITLE AND SUBTITLE **Use of Evaluation Criteria in Security Education**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Postgraduate School ,Center for Information Systems Security Studies and Research (NPS CISR),Department of Computer Science,Monterey,CA,93943**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES **3rd International Conference on Information Warfare and Security (ICIW 2008), April 2008, Omaha, Nebraska, USA, pp. 285-292**		

14. ABSTRACT

Success in information warfare will depend on resilient, reconstitutable cyber assets and the ability to assess and respond to attacks. A cornerstone of this success will be the ability of Information Assurance professionals to develop sound security requirements and determine the suitability of evaluated security products for mission-specific systems. Recognizing the pedagogical value of applying security evaluation criteria such as the Common Criteria (CC) to information security education, we recently introduced a graduate-level Computer Science course focusing on methodical security requirements engineering based on the CC. This course aims to provide students with an understanding of how security evaluation criteria can be used to specify system security objectives, derive security requirements from security objectives, establish life cycle and development processes, and provide an organizational framework for research and development. Although imperfect, the paradigmatic process of the CC provides a usable framework for in-depth study of various tasks relating to system requirements derivation and verification activities: system requirements elicitation, threat analysis, security objectives definition and security requirements expression. In-class discussions address fundamental security design principles and disciplines for information and software assurance (e.g., formal methods and life cycle management) as applied to security requirements derivation. Coverage of advanced CC topics includes high assurance evaluation, international and U.S. scheme interpretation processes, guidance for Protection Profile development, CC evaluation methodology, and composite evaluation. This paper describes the scope and design of a pilot course offering. Laboratory projects focus on the differences between security functional and assurance requirements, mock evaluation of a draft Protection Profile, examination of the interpretation process in the U.S. scheme, and development of a preliminary sketch of a Composed Assurance Package for a hypothetical composed target system that is suitable for use in operational environments requiring medium robustness. Lessons learned and planned refinements of course material and focus are also discussed.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**10**	

The Common Criteria (ISO 15408) is an internationally-recognized standard for security evaluation of IT systems (CCPS0 2006a-d). The course was formulated after our successful use of the Common Criteria (CC) as a framework for advanced student research (Nguyen 2006). The goals of the course are:

- To afford students a solid grasp of the technical aspects of defining functional and non-functional (assurance) requirements;
- To promote a more comprehensive understanding about theory and practice learned from other information assurance (IA) courses;
- To advocate the importance of having proper requirements to minimize adverse effects in the end system; and
- As a U.S. Government institution, to accord students a unique exposure to the CC whose official adoption for use in the U.S. Department of Defense (DoD) systems is promulgated in DoD directive (DoD 2003a) and instruction (DoD 2003b).

The course is designed to provide a perspective on secure systems analysis and development that can be used not only while conducting academic research, but also in the students' future positions in program management and system development. Students study security and assurance principles and disciplines in the context of security evaluation to gain an understanding of how security evaluation criteria can be used for stipulating system requirements for procurement purposes, analyzing security requirements and architectures, and providing an organizational framework for research projects. The syllabus has been derived from background materials and extensive experience in the joint development of Common Criteria protection profiles (NSA 2007) with the National Security Agency.

This paper offers some *a posteriori* observations on the course development. Background information and course prerequisites are first presented, followed by the description of the course objectives and design. A discussion of the experience with the criteria-based teaching approach is included.

2. Background

2.1 The Common Criteria

The Common Criteria (CC) provides a rigorous methodology for specifying security requirements that address protection of information from unauthorized disclosure, modification and loss of use. It is organized as three separate documents. Part 1 covers general approach and concepts (CCPSO 2006a), Part 2 contains a catalog of security functional requirements (CCPSO 2006b) and Part 3 defines security assurance requirements using a seven-level assurance scale that measures the breadth, depth, and rigor of the implementation and development process of the target system (CCPSO 2006c). The CC is intended to provide the basis for the consumers of secure products to compare and select independently evaluated products with confidence that all security requirements prescribed for the selected product have been met.

The CC is often perceived as overly complex due to its large collection of interdependent functional and assurance requirements. The requirements are expressed using a language with a defined syntax and informally-documented semantics that requires immersion in order to apply the requirements properly. A companion document to the CC is the CC Evaluation Methodology (CEM) document (CCPSO 2006d) which is used by CC evaluators to ensure that the criteria are applied correctly.

The Common Criteria Interpretations Management Board routinely issues *interpretations* concerning updates, clarifications and errata associated with the CC and CEM. Each national evaluation organization, i.e., *scheme*, can also establish scheme-specific procedures and issue its own interpretations (CCEVS 2007).

2.2 Official policies

An impetus for creating the course is to provide students the necessary skills and knowledge to help them in their development, evaluation, and deployment of IA solutions that meet the requirements stipulated in U.S. DoD Directive 8500.1 and DoD Instruction 8500.2. DoD Directive 8500.1 specifies that all IA or IA-enabled IT products used in DoD information systems must comply with the evaluation and validation requirements of the acquisition policy stated in the National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11 (CNSS 2003). This policy became effective 1 July 2002 and mandates that only validated commercial products (including cryptographic modules) can be used in national security systems. The CC is explicitly specified in DoD Instruction 8500.2 as the official evaluation framework for DoD systems.

2.3 Related IA Courses

The security requirements engineering course is part of two specialization sequences of the security track of the Computer Science curriculum at our institution. Other classes in these two sub-tracks are summarized in Table 1.

Table 1: Courses in Related Specialization Sequences

Course	*Specialization Sequence*	*Course Description*
Advanced Topics in Computer Security	Security Requirements Analysis	This class discusses academic papers on advanced topics in computer security.
Advanced Vulnerability Assessment	Computer Network Operations	This class focuses on potential vulnerabilities in networked systems.
Computer Forensics	Computer Network Operations	This class covers the fundamentals of computer forensics.
Formal Analysis of Security Protocols	Security Requirements Analysis	This class studies several cryptographic protocol analysis techniques.
Introduction to Certification and Accreditation	Security Requirements Analysis Computer Network Operations	This class provides an introduction to the Certification and Accreditation (C&A) process as applied to government systems.

The prerequisites for the course consist of one class from the core CS curriculum (Information Assurance: Introduction to Computer Security) and one class from the core sequence of the security track (Secure Computer Systems). The former offers a comprehensive overview of the concepts and issues associated with information and software assurance while the latter addresses the principles and techniques of high assurance secure system development. These prerequisites serve to ensure student familiarity with the concepts of computer and network security, information assurance, and security evaluation criteria required for successful completion of the laboratory portion of the course.

3. Course objectives

The main purpose of the course is to cultivate the student's ability to comprehend the abstract concepts and methodical processes required for security analysis and engineering of secure systems and software. Although laden with acronyms and a complex form of requirements expression, the CC approach to security requirements engineering provides a useable framework for in-depth study of various tasks relating to security requirements derivation and verification activities. The following subsections discuss these activities as applied to the course.

3.1 System requirements elicitation

The objective of this activity is to elicit from the stakeholders a consistent and achievable set of high level domain-specific requirements that the target system needs to satisfy. To achieve this goal, it is essential that the boundary for the target system be clearly established to avoid misunderstanding of the target system's objectives among users, developers and evaluators. Another challenge is to be design-neutral to avoid introducing unnecessary requirements that would drive the design toward a particular solution.

Recognizing these driving factors, the CC defines two requirement specification constructs targeted for different audiences (CCPSO 2006a). A *Protection Profile* (PP) is an implementation-independent structure used to express unambiguously security requirements that are geared towards helping end users to determine whether the target system meets their needs. A *Security Target* (ST), on the other hand, is an implementation-dependent structure used by the developers to describe security requirements specific to a target system. In both constructs, a precise description of the target system's security properties is required to minimize deviation from the security problem that the target system is intended to solve. The description also includes the target system's purpose, concept of operation, conceptual architecture, and system access policy. The course covers both types of specifications to exemplify how provision of requirements is an intrinsic part of high assurance system development.

3.2 Threat analysis

The CC addresses security in the context of protecting assets valued by both the owners and attackers. Since the countermeasures used to mitigate the risks of unauthorized access can be adversely affected by successful attacks mounted by motivated attackers, correctly characterizing the system's security environment is an essential task of requirements development. In the CC paradigm, the system's security environment is defined in terms of anticipated threats, axiomatic security assumptions about the environment, and organizational security policies. The course uses this framework to study ways to analyze inherent security threats that the target system can mitigate versus extrinsic threats for which the target system must rely upon environmental countermeasures. Organizational security policies are typically imposed to help frame the scope of the security problem, e.g., the target system shall use NIST-approved cryptographic algorithms, and thus are also included in the coursework.

3.3 Security objectives definition

Once the system's security environment is clearly defined, a set of security objectives describing the countermeasures that address all identified threats, assumptions and policies can be formed. It is important that these objectives are correctly allocated either to the target system or to the operational environment since they are used to derive the corresponding requirements. If established correctly, these objectives should collectively satisfy the target system's security goals. The course addresses the differences between the two types of objectives and the fact that with respect to CC evaluation, only the implementation of the target system's objectives is examined for correctness.

3.4 Security requirements expression

The requirements derivation process is a series of hierarchical refinement activities that starts with a description of the target system and ends with the security functional and assurance requirements that implement the established objectives. This iterative process requires demonstrable traceability via evidential material that describes pair-wise mapping of the results of the undertaken activities. If discrepancies are discovered in a particular activity, its results are invalidated and the process repeats at the previous adjacent activity. The CC defines a rich set of both functional and assurance requirements and the course uses them to foster students' understanding on different aspects of various security problems. Students need to understand that a well-engineered end system does not necessarily meet the intended operational goals if its security requirements are either erroneously or inadequately declared.

3.5 Security requirements verification

Verifying that the security requirements for a particular system are correctly expressed and verifying that the implementation of the target system is correct with respect to its requirements are distinct but similar tasks. They both need robust and reliable ways to properly assess the requirements and implementation under evaluation. The CEM describes activities to be performed by an evaluator to conduct a security evaluation of a particular target system based on the assurance requirements described in its ST. The CEM also provides guidance for the evaluator to assess if the security requirements specification of the target system, e.g., a ST, provides a sound basis for the evaluation. The course covers a sampling of the prescribed methodology to afford students contrasting views of security requirements engineering, i.e., verification versus development.

4. Course design

The course is one quarter long (eleven weeks) and combines both traditional lectures and laboratory work. There are three lectures and one laboratory session per week. Homework associated with each lecture reinforces classroom instruction and hands-on laboratory projects provide practical experience with the use of the CC from different perspectives. There are two examinations, a mid-term and a final. Each examination assesses the students' understanding on the core subjects covered in the corresponding half of the quarter. The examinations are open-book and in essay form.

The course material was initially developed based on CC Version 2.3 (CCPSO 2005a-d). The newer CC Version 3.1 (CCPSO 2006a-d) was released prior to the beginning of the pilot offering and since the security assurance requirements were significantly modified in Version 3.1, both mechanical and conceptual differences between CC Version 2.3 and Version 3.1 were incorporated into the course material. Topics covered include requirements derivation methodology with an emphasis on Protection Profile creation, security assurance requirements, especially on developmental evidence, and composite evaluation.

4.1 Lectures

The course aims to prepare students for proficiency in specifying and verification of security requirements for secure systems. The CC is utilized to provide a framework for analyzing fundamental security design principles and disciplines for information and software assurance. Specifically, the CC defines a set of functional and assurance classes of requirements that embody a number of solutions to common security functions, e.g., audit and access controls.

The lectures are structured such that upon completion of the course, the students are capable of applying the knowledge they have gained from the course work to develop CC-based security requirements and to determine suitability of evaluated security products for mission-specific use.

The lecture series starts with a “CC refresher” that reviews the CC’s general evaluation model, concepts and processes. Although the basic CC framework is taught in the two prerequisite classes, a complete review is provided so that all students have the same CC fundamentals. The next set of lectures focuses on the constructs and techniques used in the formulation of security requirements. These lectures are sequenced according to the different phases of the requirements derivation process, with an emphasis on the requirements expression phase. Both security functional and assurance requirements are studied.

Security functional requirements are used to specify testable security properties of a target system. Requirements selection is driven by security objectives and is justified with rationales that explain the choices in addition to demonstrating that the selected requirements are consistent, complete and sound. The CC groups the functional requirements into eleven functional classes, covering a wide range of security measures that a secure system could take to avert potential attacks. This catalog of requirements focuses on the following security capabilities: audit, identification and authentication, protection of communication channels, cryptography, protection of user data, security management, resource utilization and protection of the reference monitor. Hypothetical examples are used to explain the requirements and objective-based requirements selection process, e.g., from a statement of a user’s need and the corresponding security objectives, show how to select the applicable requirements that can be used to implement the objectives.

Security assurance requirements, on the other hand, constitute a way of establishing confidence in the implementation and development process of a target system. The CC also groups the assurance requirements into different assurance classes, focusing on guidance documentation, life cycle support (including configuration management), testing, vulnerability assessment, composition, and developmental evidence. The last category of requirements is of particular interest since they afford evaluators a means to gain a thorough understanding of the design and implementation of the target system. In this context the CC approach for discernment of a system’s security properties involves design decomposition and analysis of architectural soundness, both of which are fundamental and practical engineering techniques that graduate level students must master. For design decomposition, the students learn about the stepwise refinement process employed by the CC to represent the design at different levels of abstraction, with the functional description of the external interfaces as the most abstract form and the source code as the least abstract form. For architectural analysis, the students learn how to determine the soundness of the target system in terms of its internal architecture and security policy model. Class discussions cover different aspects of architectural analysis to determine how the system cannot be circumvented or changed, and how the design principles for secure system development, e.g., modularity, layering and minimization of design complexity, are applied. The study also includes discussions on the use of security policy models and correspondence mapping between the security model and the functional specification, and among different design representations.

The last part of the lecture series concentrates on a number of advanced CC topics. These include high assurance evaluation, the international and U.S. scheme interpretation process, guidance for Protection Profile development, the CC evaluation methodology, and composite evaluation.

4.2 Laboratory projects

Hands-on laboratory projects are designed to provide an in-depth study of the development of security requirements using the CC methodology. The projects focus on the differences between security functional and assurance requirements, mock evaluation of a student-developed Protection Profile, examination of the interpretation process in the U.S. scheme, and development of a preliminary sketch of a Composed Assurance Package. The projects, summarized in Table 2, are synchronized with lectures covering similar topics to strengthen students’ comprehension.

A role-playing approach is utilized to create an engaging learning environment where each student is required to work at both ends of the requirements engineering spectrum, i.e., as a developer and an evaluator. The course includes both individual and team-based projects. For team projects, all team members are required to fully understand the solution and be able to answer questions during in-class presentations.

The first project (Lab 1) focuses on the general concepts and principles of security evaluation. Part 1 of the CC provides the basis for the activity. Prior to the lab session, each student is required to submit a written report discussing three topics that the student considers worthy of group discussion during the lab session. Not surprisingly, the risks-countermeasures dichotomy, threat analysis and the unfamiliar terminology and organizational approach of the CC were popular discussion points.

Lab 2 examines the different classes of security functional requirements used in a draft PP for a multilevel print server (Lysinger 2005). The functional classes assigned to each student are jointly decided by the students working as a group. Students prepare both written reports and in-class presentations to demonstrate their proficiency in the functional classes. Specifically, they are expected to clearly articulate the problems and security objectives addressed by each class.

For Lab 3, each student acts as an evaluator and performs a mock evaluation of the draft PP. The objective of this exercise is to familiarize the students with the process undertaken by an evaluator to perform a PP evaluation. The students follow the evaluator activities described in the CEM document (CCPSO 2005d) to determine if the draft PP meets the CC requirements for a PP.

Lab 4 focuses on the security assurance requirements in the context of the draft PP. The students work together as a team to convert the applicable assurance requirements from CC Version 2.3 to Version 3.1. The premise is that the conversion work will actively engage the students in learning about these requirements instead of passively reading about them.

Lab 5 introduces the students to the CC interpretation process, an important aspect of the CC evaluation methodology. The students gain exposure to a set of specific requirements, as well as how the CC standard process is driven at the national level, i.e., the U.S. scheme. Specifically, the students study the Precedent PD-0126 (CCEVS 2006) which provides guidance on the question whether administrator-entered code invalidates the evaluatability of an evaluated product. Each student must submit a written report that includes both a summary of three topics selected from the Precedent's online discussion thread that the student considers most germane to the issue and the student's opinion regarding each topic.

Table 2: Laboratory projects

	Study Objective	*Project Type*
Lab 1	General evaluation model	Individual
Lab 2	Security Functional Requirements	Team
Lab 3	Evaluation Methodology	Individual
Lab 4	Security Assurance Requirements	Team
Lab 5	Interpretation Process	Individual
Lab 6	Composed Assurance Package	Individual

The last project (Lab 6) is a capstone activity since the Composed Assurance Package development process requires the students to apply their understanding of the course material to derive a set of assurance requirements (i.e., an assurance package in CC parlance) for a hypothetical target system that is composed of multiple evaluated components and is suitable for use in environments requiring medium robustness. *Robustness* is a metric used in the U.S. evaluation scheme to measure the target system's ability to protect itself and its resources. Robustness is defined in terms of the strength of the target system's security mechanisms and its level of assurance (NSA 2002). There are three levels of robustness: high, medium, and basic. The required robustness level of a target system is calculated based on the value of the data that the system must protect and the perceived threats that the system must mitigate. Medium robustness is resistant to sophisticated threats from an organized effort and thus can be used to protect medium-value data (NSA 2002).

5. Discussion

The course follows a learning-by-doing pedagogical approach and uses the CC as a teaching apparatus to promote critical thinking skills that employers increasingly seek in their employees. It was envisioned that this combination would foster student's active exploration of core subjects in security requirements engineering. The result of the pilot offering of the course shows that this vision was not fully realized due to a number of technical and conceptual barriers.

5.1 Students readiness

Although core security principles and key CC concepts are taught in the prerequisite courses, it quickly became apparent that the students were not prepared to undertake the tasks required by the course. Even with the "CC refresher" lectures, it was still challenging for the students to comprehend the relevant topics since the CC learning curve was steep for the students. The time elapsed between when the students took the prerequisite classes and this course was also a factor since the information learned from those classes was often forgotten due to a lack of hands-on reinforcement immediately afterward.

5.2 Class dynamics

The course is an elective and thus the class size was small, which induced both positive and negative effects on class dynamics. Since fifteen percent of the final grade came from class participation, it was important for the students to speak in class and the class's small size helped the "quiet ones" to be less intimidated. However, the small size also afforded the talkative students more opportunities to divert class attention from the issues under discussion. Some digressions stemmed from the CC-heavy reading assignments. These tangents often sidetracked the study plan of the day and sometimes resulted in additional homework assignments to ensure that the students attained the lesson's goals.

In-class debates also had pros and cons. For straightforward topics, the students were more engaged and the debates helped to reinforce the lessons. However, for complex issues, some students only participated passively which unwittingly encouraged the more opinionated students to polarize the discussions with subjective beliefs. Most students were able to recognize and ignore these assertions.

5.3 Course contents

The course instructor faced a number of challenges concerning what to teach and how best to teach highly abstract concepts. The desire to nurture students to appreciate the value of rigorously-developed secure system and software led to a CC-immersed workload that was too intensive for the students. The strict language and constructs required by the CC also impeded the students' ability to satisfactorily complete class assignments.

The six laboratory projects were devised to increase in complexity as the course progressed, ranging from the mechanistic utilization of the CC methodology to the construction of new requirements for a composed system. The objective was to foster critical and logical thinking skills that can synthesize the inherently “dry” security evaluation concepts to promote better understanding and long-term retention of the learned principles and techniques. This expectation was too optimistic since the students were unable to internalize the CC methodology sufficiently to apply it to even small practical problems in security requirements engineering.

6. Conclusion

This paper describes the goal and implementation of a new course that uses security evaluation criteria to teach security requirements engineering concepts and mechanisms. The course seeks to help students acquire practical concepts and hone their analytical thinking skills. The course’s objective aligns with our institution’s mission to prepare students for professions that require graduates to critically and objectively assess complex and sometimes perplexing issues.

The pilot offering was not designed to appeal to students looking for an easy elective. This resulted in less favorable feedback from students, particularly those with biases against the CC approach. Hence, the lectures and laboratory projects are being overhauled to be less CC centric. We are also exploring ways to incorporate relevant aspects from other research efforts (Haley 2006) to better increase students’ awareness that good software and system security starts with having correct requirements.

Acknowledgements

This work was sponsored by the Office of Naval Research. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsoring organization.

References

Cheng, B. H. C. and Atlee, J. M. (2007) "Research Directions in Requirements Engineering," in *Future of Software Engineering 2007, FOSE’07*, pp. 285-303.

Committee on National Security Systems (2003) “National Security Telecommunications and Information Systems Security Policy (NSTISSP) No. 11,” June.

Common Criteria Evaluation and Validation Scheme (2006) ”PD-0126: Administrator-entered Code Used To Meet SFRs,” http://www.niap-ccevs.org/cc-scheme/PD/0126.html.

Common Criteria Evaluation and Validation Scheme (2007) ”The Interpretation Process,” http://www.niap-ccevs.org/cc-scheme/interps-process.cfm.

Common Criteria Project Sponsoring Organizations (2005a) “Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model,” CCIMB-2005-08-001, Version 2.3, August.

Common Criteria Project Sponsoring Organizations (2005b) “Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements,” CCIMB-2005-08-002, Version 2.3, August.

Common Criteria Project Sponsoring Organizations (2005c) “Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance requirements,” CCIMB-2005-08-003, Version 2.3, August.

Common Criteria Project Sponsoring Organizations (2005d) “Common Criteria for Information Technology Security Evaluation, Evaluation methodology,” CCIMB-2005-08-004, Version 2.3, August.

Common Criteria Project Sponsoring Organizations (2006a) “Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model,” CCIMB-2006-09-001, Version 3.1 Revision 1, September.

Common Criteria Project Sponsoring Organizations (2006b) “Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements,” CCIMB-2006-09-002, Version 3.1 Revision 1, September.

Common Criteria Project Sponsoring Organizations (2006c) "Common Criteria for Information Technology Security Evaluation, Part 3: Security assurance requirements," CCIMB-2006-09-003, Version 3.1 Revision 1, September.

Common Criteria Project Sponsoring Organizations (2006d) "Common Criteria for Information Technology Security Evaluation, Evaluation methodology," CCIMB-2006-09-004, Version 3.1 Revision 1, September.

Haley, C. B., Moffett, J. D., Laney R., Nuseibeh, B. (2006) "A framework for security requirements engineering," in *Proceedings of the 2006 International Workshop on Software Engineering for Secure Systems*, Shanghai, China, pp 35-42.

Lysinger III, J. E. (2005) "Multilevel Print Server Requirements for DoN Application," Masters Thesis, Naval Postgraduate School, Monterey, California, USA.

National Security Agency (2007) "U.S. Government Protection Profile for Separation Kernels in Environments Requiring High Robustness," Version 1.03, 29 June 2007.

National Security Agency (2002) "Information Assurance Technical Framework," Chapter 4, Release 3.1, September.

Nguyen, T. D. and Irvine, C. E. (2006) "Utilizing the Common Criteria for Advanced Student Research Projects," in *IFIP International Federation for Information Processing, Volume 201, Security and Privacy in Dynamic Environments*, eds. Fischer-Hubner, S., Rannenberg, K., Yngstrom, L., Lindskog, S., (Boston: Springer), pp. 317-328.

Nuseibeh, B. and Easterbrook, S. (2000) "Requirements engineering: a roadmap," in *Proceedings of the Conference on the Future of Software Engineering*, Limerick, Ireland, pp 35–46.

United States of America Department of Defense (2003a) "Department of Defense Directive Number 8500.1," October 24, 2002, Certified Current as of November 21, 2003.

United States of America Department of Defense (2003b) "Department of Defense Instruction Number 8500.2," February 6, 2003.

***See* reference**

CCEVS – see Common Criteria Evaluation and Validation Scheme
CCPSO – see Common Criteria Project Sponsoring Organizations
CNSS – see Committee on National Security Systems
DoD – see United States of America Department of Defense
NSA – see National Security Agency